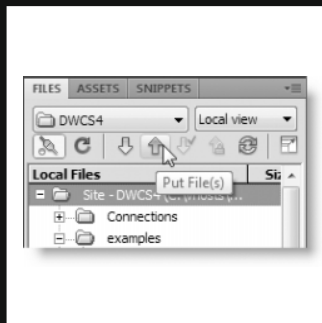


20 DEPLOYING YOUR SITE ON THE INTERNET



You've built your site and created your database, and now it's time to go live. If you have structured your Dreamweaver site as an exact copy of the site you intend to deploy on the Internet, going live is a pretty straightforward process. Assuming you have defined your remote server as described in Chapter 2, you can deploy your web pages a number of ways through the Dreamweaver interface. Probably the most common way to do so is through the Files panel. Select the file(s) you want to upload, and click the Put button. You can also upload the current page by pressing Ctrl+Shift+U/Shift+Cmd+U, or selecting Put from the File management menu on the Document toolbar (see Figure 1-25 in Chapter 1). You can even right-click files in the Results panel and select Put from the context menu. However, Dreamweaver is not capable of uploading the content of a MySQL database; and there are several other issues with deploying a site that sometimes cause confusion, so this chapter tries to clear up any doubts you might have.

In this chapter, you'll learn about the following:

- Deciding which files to upload
- Preventing development assets from being uploaded
- Exporting a MySQL database to a SQL file
- Using a SQL file to transfer a MySQL database to another server
- Adapting connection files to work with your live MySQL server

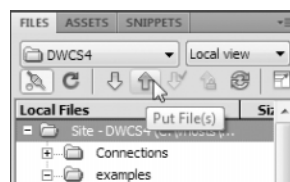
Uploading your web pages

The key to uploading your website successfully lies in registering the details of your remote server correctly in the Dreamweaver Site Definition dialog box, as described in Chapter 2. Dreamweaver considers the site root to be whatever value you entered in the Host directory field of the Remote Info category (see Figure 2-6); and it uploads files using the same folder hierarchy as in the Files panel. Usually, there is no need to create new folders on the remote site; Dreamweaver does it automatically when you upload the contents of the folder.

The following sections describe how to upload files to a remote server. However, apart from designating individual files and folders to be excluded from batch uploading—a technique Dreamweaver calls **cloaking**, the same techniques apply equally to downloading files. To download, use the Get button instead of the Put button.

Uploading a whole site

The easiest way to upload a whole site is to select the site's root folder in the Files panel, and click the Put button, as shown in the following screenshot:



Before it goes ahead, Dreamweaver displays an alert asking you to confirm this operation. If you click OK, it takes you at your word and uploads everything within the site. However, be warned that Dreamweaver's FTP engine is unlikely to win any Olympic gold medals. I find it adequate for most site maintenance, but it can be excruciatingly slow if you have a lot of files. For example, a typical WordPress blog (<http://wordpress.org/>) contains more than 750 files in 175 folders. Although Dreamweaver uploads the files in the background, you cannot switch the Files panel to work on a different site until the upload process has completed. There's also no way to resume the upload if the connection is dropped. When uploading a site of this size, I find it much faster to use a third-party FTP program, such as FileZilla, a free open source program for Windows, Mac, and Linux (<http://filezilla-project.org>), or CuteFTP, a commercial program for Windows and Mac (http://www.cuteftp.com/products/ftp_clients.aspx).

For a few dozen files, uploading the whole site through Dreamweaver is no problem, but you should still restrict the files that are transferred by cloaking anything that's not required on the remote server.

Cloaking files not required on the remote server

Cloaking prevents files from being uploaded to your remote server when uploading the entire site or using Dreamweaver's site synchronization feature (described later in this chapter). You can cloak files in two ways: you can apply cloaking automatically to all files that have a particular file name extension, or you can designate individual files or folders.

To enable automatic cloaking for specific file name extensions, open the Site Definition dialog box, and select the Cloaking category (see Figure 20-1). The quick and easy way to do this is by right-clicking any file or folder in the Files panel and then selecting Cloaking ► Settings from the context menu.

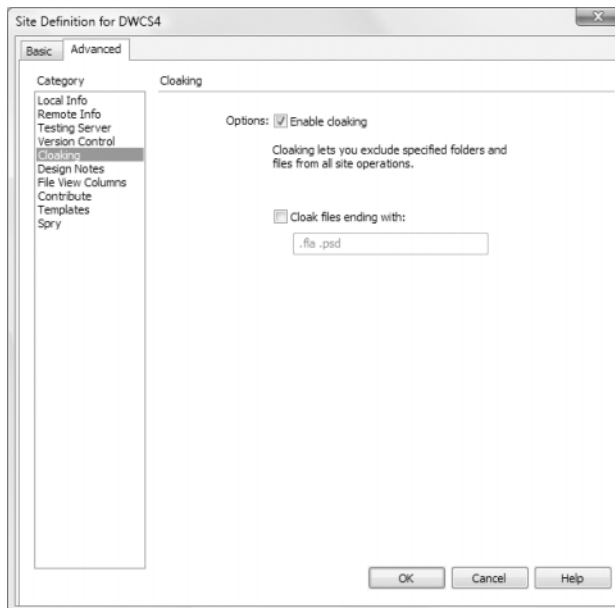
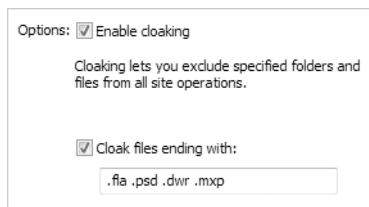


Figure 20-1. You can enable automatic cloaking of specified file types through the Site Definition dialog box.

To designate the types of files to be cloaked, select the checkbox labeled *Cloak files ending with*. This enables the text field below, where you enter all the file name extensions that you want to exclude when uploading or synchronizing the site. As you can see in Figure 20-1, *.fla* and *.psd* are already listed. These are Flash and Photoshop work files that store all the layers, symbols, and other information about a Flash movie or image. Neither should ever be deployed on a website.

Sites created with earlier versions of Dreamweaver and migrated to Dreamweaver CS4 list *.png* instead of *.psd*. This is the file name extension for work files created by Fireworks. However, it is also the file name extension for images stored in the Portable Networks Graphic (<http://www.libpng.org/pub/png/>) format supported by all modern browsers. So if your site uses *.png* images, you might want to remove *.png* from the list.



There should be a single space between each file name extension, and the extension itself should begin with a period. So, for example, if you want to exclude saved queries for Find and Replace (*.dwr* files) and Dreamweaver extensions (*.mxp* files), you need to edit the list as shown in the screenshot alongside:

*Templates (as distinct from child pages created from a template) should never be uploaded to the remote server. If you use templates, add *.dwt* and *.dwt.php* to this list.*

When you click OK to save the changes, Dreamweaver displays an alert telling you that it needs to rebuild the site cache. Click OK again, and after the site cache has been rebuilt, files belonging to the specified types are displayed in the Files panel with a red diagonal stripe through the icon alongside the file name, as shown in Figure 20-2. Rebuilding the cache can take a long time on a large site, so this is not an option you should change frequently. It's best done when you first define the site.

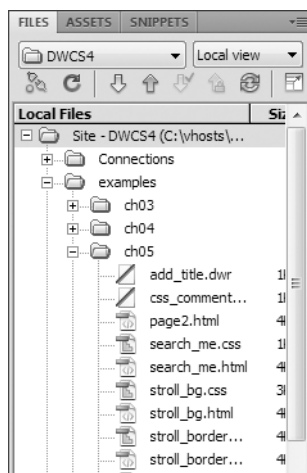


Figure 20-2.
A red diagonal stripe through the icon alongside a file or folder indicates it has been cloaked.

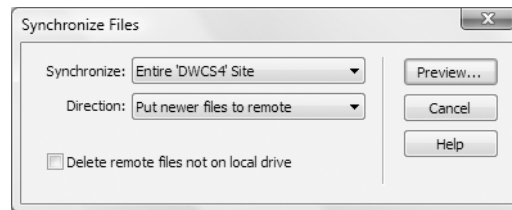
To cloak individual files or folders, right-click the file or folder in the Files panel, and select Cloaking ► Cloak from the context menu.

Cloaking excludes files and folders when you select the entire site to be uploaded or when synchronizing the site. However, you can override cloaking temporarily by selecting cloaked files or folders and clicking the Put button.

To turn off cloaking for individual files and folders, right-click the file(s) or folder(s) in the Files panel, and select Cloaking ► Uncloak. However, you cannot use this method to uncloak files that have been cloaked automatically through their file name extension. Automatic cloaking is an all-or-nothing option: all files that use the same extension are affected. Equally, selecting Cloaking ► Uncloak All from the context menu in the Files panel is a blanket operation; it uncloaks every file in the site.

Synchronizing a site

By default, Dreamweaver maintains information about the local and remote versions of the site to enable them to be synchronized. To synchronize a site, select Site ► Synchronize Sitewide. Alternatively, select Synchronize from the context menu in the Files panel. This opens the Synchronize Files dialog box, as shown here:



The Synchronize drop-down menu has the following two options, both of which are self-explanatory:

- Entire '*Sitename*' Site
- Selected Local Files Only

The Direction drop-down menu has the following three options:

- Put newer files to remote: This updates the remote site with files that have been changed in Dreamweaver.
- Get newer files from remote: This updates your local version with files that have been updated on the remote server. Normally, this applies only if more than one person is responsible for managing the site or if you use more than one computer to maintain a site.
- Get and Put newer files: This looks for the latest versions on both the remote server and your local version and transfers them accordingly.

The options offered by the checkbox at the bottom of the Synchronize Files dialog box depend on your choice for Direction. When putting newer files to the remote server, select the checkbox to delete files that don't exist in your local version. When getting newer files from the remote server, select the checkbox to delete local files that don't exist on the remote server. The checkbox is grayed out if you set Direction to Get and Put newer files. Although deleting files sounds potentially dangerous, you get a chance to review Dreamweaver's decision for each file before committing yourself.

After making your selections in the Synchronize Files dialog box, click the Preview button. Dreamweaver connects to the remote server and checks the status of the remote files. This can take some time, depending on the size of the site. When Dreamweaver has gathered the synchronization information, it displays the Synchronize dialog box (see Figure 20-3). This lists all the files that are out of sync. The buttons at the bottom left of the dialog box give you the opportunity to override Dreamweaver's proposed actions or resolve issues that Dreamweaver can't handle. The dialog box is resizable to make it easier to read messages in the Status field. The Compare button requires a third-party file comparison utility to be set up in the File Compare category of the Preferences panel, as described in Chapter 2.

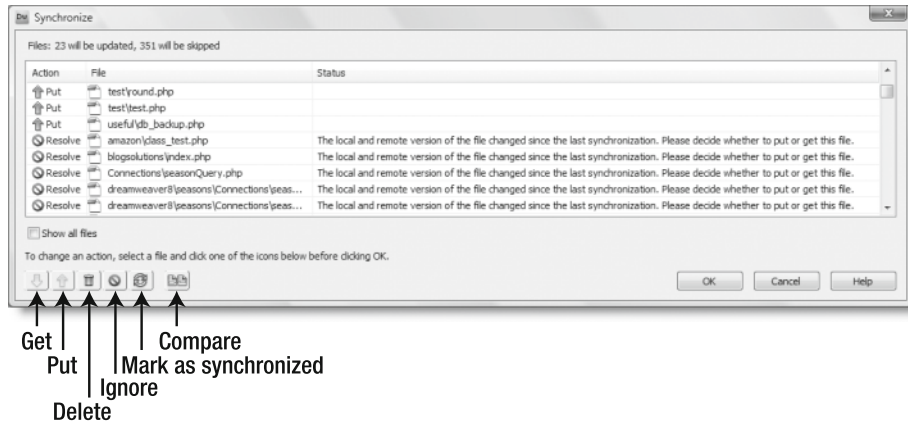


Figure 20-3. The Synchronize dialog box gives you the opportunity to override Dreamweaver's decisions.

After making any changes to the proposed operations, click OK, and sit back while Dreamweaver carries out your instructions. You can click the Hide button on the Background File Activity dialog box and continue working in Dreamweaver. However, you cannot switch to another site until the transfer has completed.

The file synchronization feature creates a file called `dwsync.xml` for each folder in your site. This XML file contains details of the local and remote timestamps of each file in the folder, and it is stored in a hidden folder called `_notes` (see <http://www.adobe.com/go/kb400972> for a detailed explanation). For the system to work accurately, both your local computer and remote server need to create accurate timestamps. Problems inevitably arise if your computer clock is out of sync with the one on the remote server.

The greatest weakness of file synchronization affects everyone in countries that observe daylight saving time. Twice a year, when the clocks change, Dreamweaver displays inaccurate warnings that you're about to replace a newer file on the remote server. This happens even when you use normal Put operations, because Dreamweaver checks the information in `dwsync.xml` each time a file is transferred. If in doubt, use File Compare, as described in Chapter 2.

You can find synchronization information about an individual file by right-clicking the file name in the Files panel and selecting Display Synchronize Information from the context menu. This displays a panel of information similar to that shown in Figure 20-4.

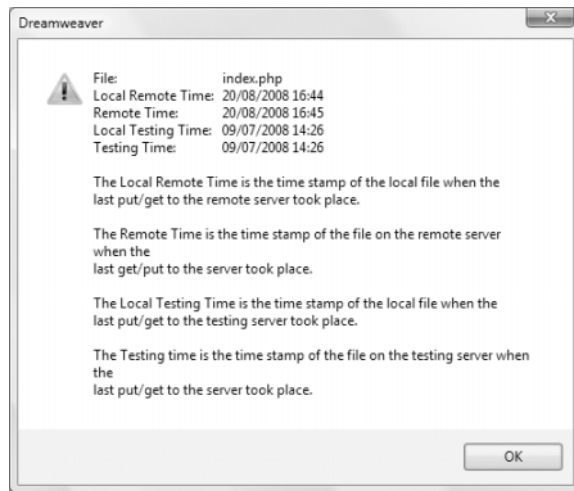
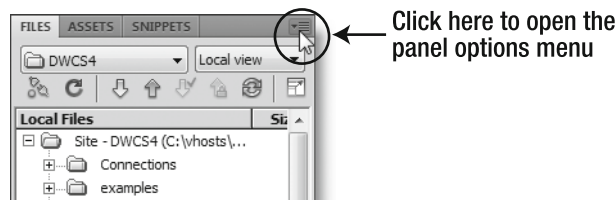


Figure 20-4. You can display synchronization information for an individual file.

Selecting recently modified files

If you find synchronization unreliable or too time-consuming, the Files panel has a little known feature that selects recently modified files. To access this feature, open the Files panel options menu by clicking the top-right of the panel, as shown in the following screenshot:



In the panel options menu, select **Edit** ► **Select Recently Modified**. Alternatively, right-click any file or folder in the Files panel, and select **Select** ► **Recently Modified** from the context menu. This opens the dialog box shown in Figure 20-5. After setting your selection criteria, click **OK**. Dreamweaver selects all matching files in the Files panel, which are then ready for you to upload by clicking the **Put** button, or for you to perform any other batch operation. To select all files created or modified during the current day, select the second radio button, and set both dates to today's date.

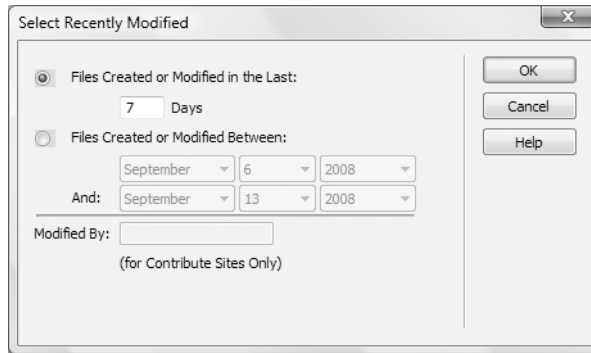
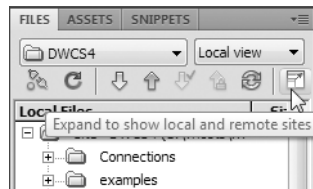


Figure 20-5. You can select files that have been created or modified in recent days or within a specific range of dates.

The Results panel also lets you upload files that have just been modified by a **Find and Replace** operation. Right-click the file name in the Results panel, and select **Put** from the context menu.

Viewing the local and remote sites side by side

The drop-down menu at the top right of the Files panel lets you select only one view of your site at a time: **Local view**, **Remote view**, **Testing server**, or **Repository view** (this last option requires a Subversion repository to be set up in your site definition, as described in Chapter 2). However, it can be useful to display two views side by side. To do so, click the **Expand/Collapse** button at the top right of the Files panel, as shown here:



By default, this expands the Files panel to fill the application window, as shown in Figure 20-6. However, if you have converted the Files panel to a floating panel, it remains floating and resizable. In both cases, the local view remains on the right side of the panel, while the left side opens to display the remote view. If no files are displayed in the left panel, Dreamweaver prompts you to connect to the remote server. If you prefer to show the local files on the left, you can reverse the default positions in the Site category of the Preferences panel (Edit ► Preferences, or Dreamweaver ► Preferences on a Mac).

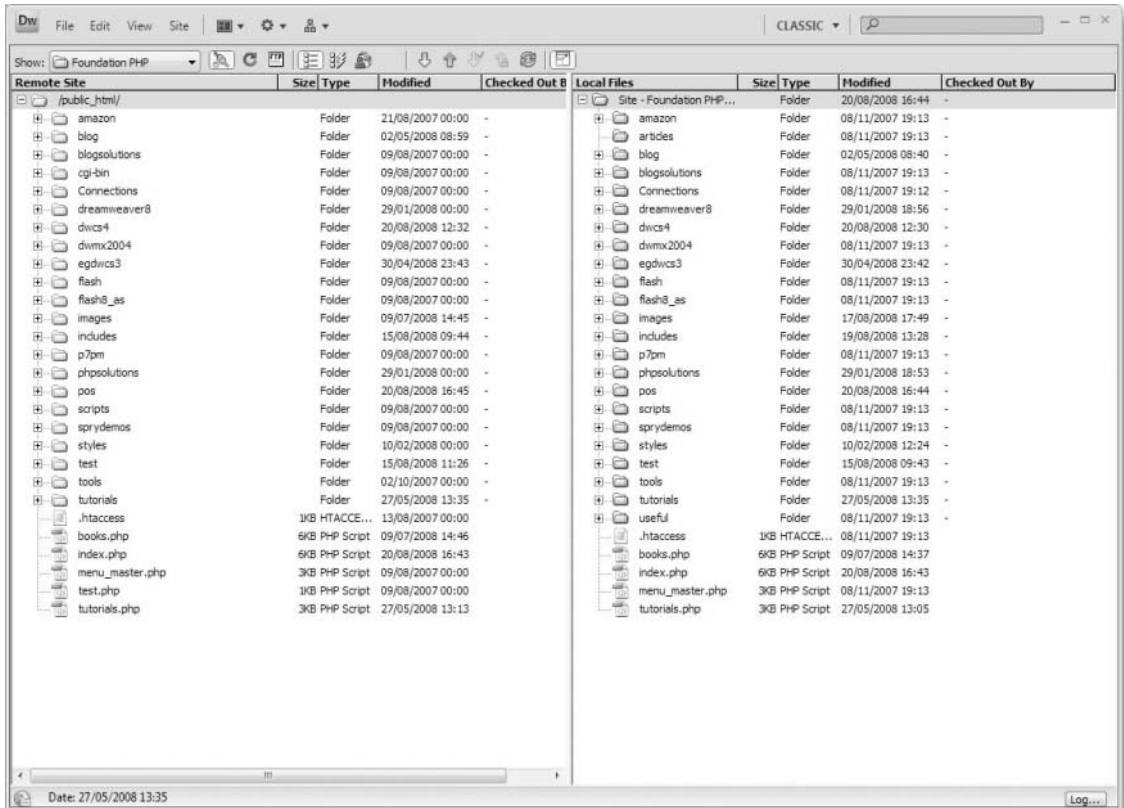


Figure 20-6. When expanded, the Files panel shows details of the remote files alongside the local ones.

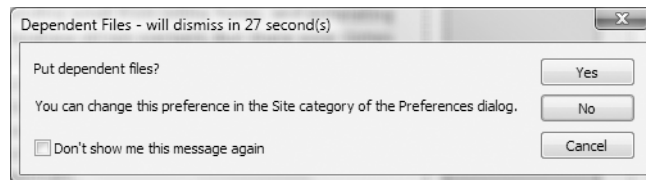
You can use this expanded view to drag and drop local files into the remote site, and vice versa, but it's a tricky process because you need to drop the files on the target folder icon. It's easy to miss the correct icon, destroying the file hierarchy of your site. Instead of dragging and dropping, expand the folders to highlight the files you want to transfer, and use the Put and Get buttons to perform the actual transfer.

You can also inspect the testing server and repository view in the expanded Files panel by clicking the buttons to the left of the Get button.

To collapse the Files panel to its normal size, click the same button you used to expand it.

Uploading dependent files

Most web pages have a number of dependent files: style sheets, external JavaScript files, images, and so on. Dreamweaver presents you with the following dialog box whenever you click the Put or Get button, asking whether you want to transfer dependent files at the same time.



It's important to realize that the default button on this dialog box is No. If you press Enter/Return or do nothing for 30 seconds, the dialog box closes and transfers only the selected file(s). To transfer dependent files, you must click the Yes button before the dialog box closes automatically. If you don't want to be asked each time you perform a transfer operation, select the checkbox labeled Don't show me this message again.

You can change the dependent file options in the Site category in the Preferences panel (Edit ► Preferences; Dreamweaver ► Preferences on a Mac). Dreamweaver treats dependent files separately for put and get operations, so you can reenable the automatic prompt by selecting the appropriate checkbox in the Preferences panel. You can also change the number of seconds the prompt remains on the screen or prevent it from closing automatically. If you uncheck Select default action after *n* seconds, the Dependent Files dialog box remains onscreen until you click one of the buttons. More importantly, the default button changes from No to Yes.

Deciding whether to let Dreamweaver transfer dependent files automatically is a matter of preference. Experienced developers usually want to know exactly what is being transferred, so they disable the Dependent Files dialog box. If you do likewise, don't forget to upload the following categories of files:

- Images
- External style sheets
- External JavaScript files
- Assets for Spry and other web widgets
- SWF (Flash and Flex) support files in the Scripts folder

One of the most frequently asked questions in Dreamweaver online forums is “Why does my Flash movie, Spry accordion, or some other feature work fine locally but not on my remote server?” The answer is almost invariably that a dependent file hasn't been uploaded. Another explanation is that the file path to the dependent file is incorrect. That's usually because the site has been incorrectly defined.

Dreamweaver cannot detect dependent files that are linked dynamically through JavaScript or PHP. You must upload these independently.

Transferring database tables

MySQL doesn't store your database in a single file that you can simply upload to your web-site. Even if you find the right files, you're likely to damage them unless the MySQL server is turned off. Anyway, most hosting companies won't permit you to upload the raw files, because it would also involve shutting down their server, causing a great deal of inconvenience for everyone.

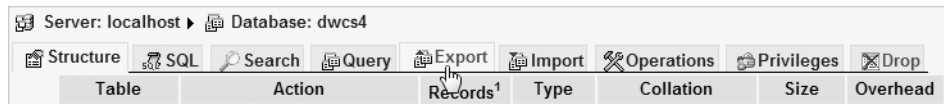
Nevertheless, moving a database from one server to another is very easy with a graphic front end to MySQL. I'll show you how to do it with phpMyAdmin, since most hosting companies provide phpMyAdmin as part of their package. If you want to use a different front end, consult the program's documentation. However, for security reasons, most hosting companies do not permit direct connection to the MySQL server, so you normally need to use the host's installation of phpMyAdmin to import the data. Navicat (<http://www.navicat.com/>) gets around this issue by transferring the data through the web server on port 80.

Transferring a database or database table with phpMyAdmin involves creating a backup **dump** of the data and loading it into the other database. The dump is a text file that contains all the necessary Structured Query Language (SQL) commands to populate an individual table or even an entire database elsewhere.

Creating a backup with phpMyAdmin

These instructions show you how to back up an entire database. You can also back up individual tables in the same way by selecting the tables in step 4.

1. Launch phpMyAdmin, and select the database you want to back up from the list or drop-down menu in the navigation frame.
2. When the database details have loaded into the main frame, select Export from the tabs along the top of the screen, as shown here:



3. The rather fearsome-looking screen shown in Figure 20-7 opens. In spite of all the options, you need to concern yourself with only a few.

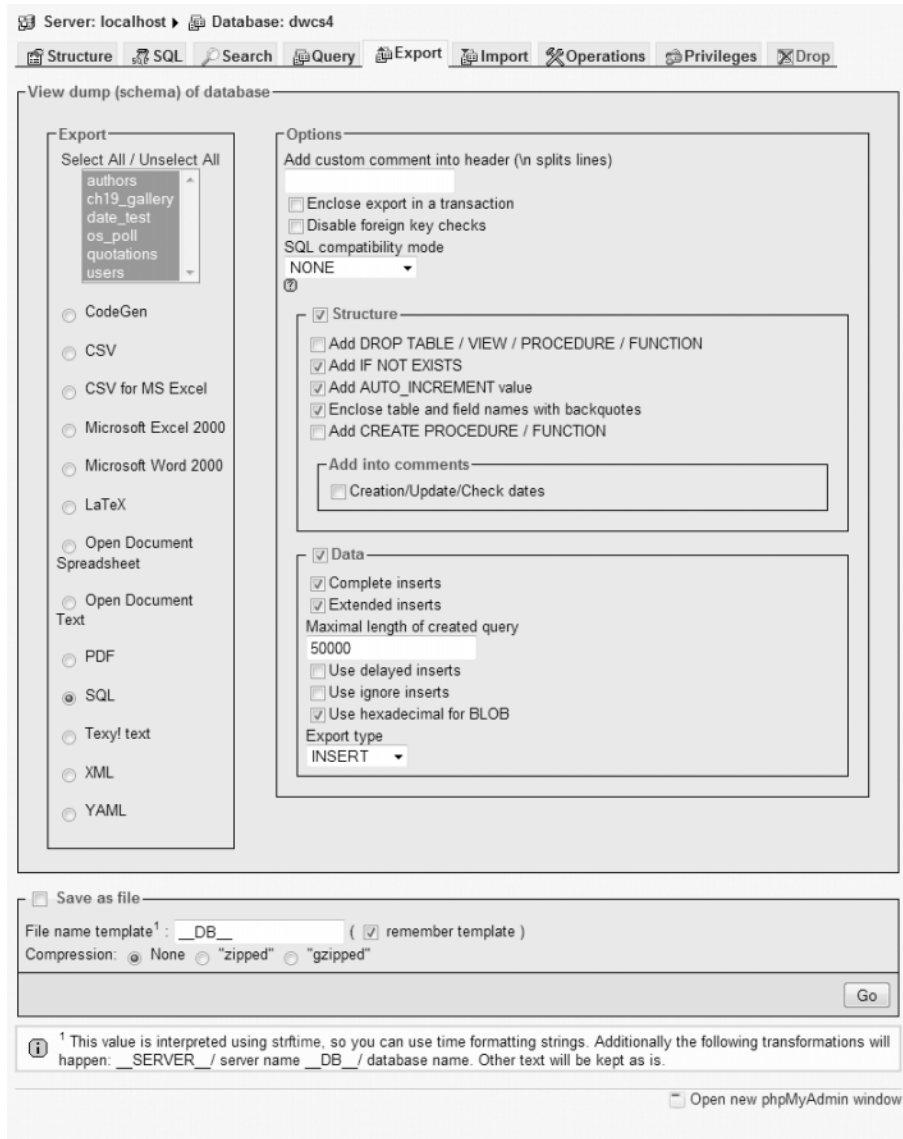


Figure 20-7. phpMyAdmin offers a wide range of choices when exporting data from MySQL.

4. The Export section on the left of the screen lists all the tables in your database. Click Select All, and leave the radio buttons on the default SQL. Alternatively, Ctrl-click/Cmd-click to select the tables you want to export.
5. If the database has *never* been transferred to the other server before, the only option that you need to set on the right side of the screen is the drop-down menu labeled SQL compatibility mode. The setting depends on the version of

MySQL on the other server (only the first two numbers, such as 4.0, 4.1, or 5.0, are important):

- If the other server is running the same version of MySQL, choose NONE.
 - If you are transferring between MySQL 4.1 and MySQL 5.0 (in either direction), choose NONE.
 - If the other server is running MySQL 4.0, choose MYSQL40.
6. If the database has *already* been transferred on a previous occasion, select Add DROP TABLE / VIEW / PROCEDURE / FUNCTION in the Structure section. The existing contents of each table are dropped and are replaced with the data in the backup file.
 7. Put a check mark in the box alongside Save as file at the bottom of the screen. The default setting in File name template is __DB__, which automatically gives the backup file the same name as your database. So, in this case, it will become dwcs4.sql. If you add anything after the final double underscore, phpMyAdmin will add this to the name. For instance, you might want to indicate the date of the backup, so you could add 20090704 for a backup made on July 4, 2009. The file would then be named dwcs420090704.sql.

Loading data from a backup file

Once you have created a .sql file containing the data exported from a database, you can transfer it to another MySQL server using the version of phpMyAdmin on the remote server.

1. If a database of the same name doesn't already exist on the target server, create the database, but don't create any tables.
2. Launch the version of phpMyAdmin used by the target server, and select the database you plan to transfer the data to. Click the Import tab in the main frame (on versions of phpMyAdmin earlier than 2.7.0, click the SQL tab instead).
3. Use the Browse button to locate the SQL file on your local computer, and click Go. That's it!

Because phpMyAdmin uses PHP to upload the file, the maximum size of any backup is normally limited to 2MB, which is the default maximum size for any file upload. If you are transferring a very large database, use the phpMyAdmin Export and Import tabs to back up and transfer individual tables. Alternatively, contact your hosting company for advice on transferring your database.

Configuring the remote MySQL connection

Many hosting companies locate MySQL and the web server on the same computer. Consequently, the host name PHP uses to connect to the database remains localhost. If you set up your local testing environment to use the same database name, user account,

and password as your remote server, all you need to do is upload the Connections folder and its contents, and everything should work the same as on your local computer.

However, some hosting companies locate MySQL on a separate computer and give you an IP address or domain name that must be used instead of localhost. Unfortunately, Dreamweaver does not have the option to specify different connection details for local testing and remote deployment. The simplest way to get around this problem is to create different copies of the connection files for the local and remote connection. Once you have uploaded the connection file(s) to your remote server, cloak the local connection file(s) in your Dreamweaver site to prevent accidentally overwriting the remote versions (cloaking was described earlier in this chapter).

Figure 20-8 shows the contents of the connection file for the connAdmin user account used in the examples for this book. As you can see, the file name and the variables used in the file are all based on the name of the user account. The variable names are all self-explanatory.



```

1 <?php
2 # FileName="Connection_php_mysql.htm"
3 # Type="MYSQL"
4 # HTTP="true"
5 $hostname_connAdmin = "localhost";
6 $database_connAdmin = "dwcs4";
7 $username_connAdmin = "cs4admin";
8 $password_connAdmin = "humpy";
9 $connAdmin = mysql_pconnect($hostname_connAdmin, $username_connAdmin, $password_connAdmin) or trigger_error(
10 mysql_error(), E_USER_ERROR);
11 ?>

```

Figure 20-8. The MySQL connection file contains the login details for the MySQL user account.

To create a connection file for your remote server, use the following instructions:

1. Save a copy of the connection file as *filename_local.php*.
2. Close the copy, and edit the original connection file, replacing localhost (shown on line 5 of Figure 20-8) with the IP address or domain name of the server where your MySQL account is located (use the MySQL domain name, *not* your own domain name).
3. Make any other changes necessary to the database, username, and password.
4. Save the page, and upload it to your remote server.
5. Delete the local copy of the page you have just uploaded.
6. Rename *filename_local.php* to restore its original name.
7. Right-click the renamed local file, and select Cloaking ► Cloak from the context menu.

You now have different versions of the connection file on your local computer and remote server. This enables you to continue testing locally with your local MySQL settings, and the files should work seamlessly with the remote settings when they're uploaded. However, it's important not to select the option to upload dependent files, because this will overwrite

the remote connection file, even though it has been cloaked. Cloaking excludes files only when using file synchronization.

When editing connection files, make sure you don't add any whitespace or newline characters outside the PHP tags, because this will prevent sessions from being created and server behaviors from redirecting to other pages. Also, make sure you don't select Include Unicode Signature (BOM) in the page properties (see "Avoiding the 'headers already sent' error" in Chapter 12).

Chapter review

This brief look at deploying your website on the Internet brings to an end this marathon journey through working with Dreamweaver CS4. I hope you have found it enjoyable and instructive. In spite of the length of this book, I have not been able to cover every aspect of Dreamweaver, CSS, Ajax, and PHP. In the short history of the Internet, the technology of website design has grown at a breathtaking pace—almost as rapid as that of the Internet itself. Dreamweaver will do a lot of the hard work for you, but your success in creating websites that stand out from among the crowd depends on mastering a range of skills, or perhaps combining your own strengths with those of experts in other aspects of web development.

Good luck, but above all have fun.